

DOCUMENT REVISION SUPPORT PROGRAM AND COMPUTER
READABLE MEDIUM ON WHICH THE SUPPORT PROGRAM IS
RECORDED AND DOCUMENT REVISION SUPPORT DEVICE

BACKGROUND OF THE INVENTION

5 The present invention is directed to a system for supporting a document revision process.

 There has been a demand to keep document revision histories for future use. Conventionally, revision histories have been kept in the form of paper documents by simply jotting down changes directly onto the documents, or
10 a copy of a revised document has been made and kept on a generation-by-generation basis, or a revised document file for each generation has been saved using a file function of a word processor and the like. As a more advanced means of
15 keeping document revision histories, a technique for retaining a differential between an original document and a revised document has been employed. One of the most prevalently used means at present is a revision record function and format for saving revision histories provided
20 by application software Microsoft Word ®.

 The aforementioned conventional means of keeping revision histories are in some ways inconvenient. For example, when keeping revision histories in the form of paper documents, not only is proper maintenance of such
25 paper documents difficult but also substantial storage space is needed. Even with the use of electronic files of word processors and the like, there is a limit to what one can do to control document revision histories, such as recording the generation of several revisions at an
30 optional time to compare the same with the other recorded generation of revisions, retrieving a document of a previous generation to replace a document of the current generation or making further changes to a revision of a previous generation. Thus, the conventional means is
35 inadequate and cannot meet the need for strict control of revision histories in revising laws, ordinances and special documents such as legal documents, patent specifications, industrial codes, documents relating to financial matters,

specifications, instruction and other manuals, office regulations, bylaws, and the like. In other words, it requires the following functions that cannot be implemented by the conventional means to meet the aforementioned needs:

- 5 1) To automatically generate from a revision record a "revision description" that specifically describes the contents of a revision made at certain point in time;
- 2) To selectively cancel a specific revision among multiple revisions that are intricately related to one another;
- 10 3) To concurrently perform a multiplicity of revision operations on the same document;
- 4) To control multiple revisions independently of one another and selectively reflect the contents of the revisions in the original document;
- 15 5) To automatically generate a cross reference table comparing a document before a revision with the document after the revision; and
- 6) To inspect the content of an optional revision in a document.

20 The above functions are essential especially when revising laws and ordinances and the like, and the conventionally available technique not only requires enormous time and manpower but is also prone to errors.

SUMMARY OF THE INVENTION

25 An object of a document revision support system of the present invention is to obviate the aforementioned problems of the prior art. Therefore, the present invention provides a means for saving documents in an XML format and also keeping whole revision histories of the documents in the XML documents, and achieves the above
30 object by providing such an XML document processing system.

More specifically, the present invention provides a means for adding attributes and elements for saving document revision histories to schema (document type
35 definition DTD) in XML prescribed for saving documents, thereby recording and saving not only documents but also the documents' revision histories including the order in which the revisions were made. In other words, the present

invention provides a means for referencing and controlling a differential for each structure of an XML document. Further, the present invention realizes a system for automatically generating, by using recorded and saved revision history data, a "revision description" that specifically describes the contents of a revision made at a certain point in time, selectively canceling a specific revision of multiple revisions that are intricately related to one another, concurrently performing a multiplicity of revision operations on the same document, controlling multiple revisions independently of one another and selectively reflecting the contents of the revisions on the original document, automatically generating a cross reference table comparing a document before a revision with the document after the revision and inspecting the content of an optional revision in a document.

A document revision support system of the present invention is able to support operations for revising various documents whose revision histories need to be strictly controlled and is especially effective in supporting an operation for revising laws and ordinances and the like, which require a complex revision operation in strict compliance with a prescribed format.

In revising laws and ordinances and the like, it is necessary to record as revision histories the contents of revisions of a document as well as the order in which the revisions were made, to generate on the basis of the thus recorded histories a "revision description (revised law or revised ordinance)" succinctly describing the contents of a revision at a certain point in time and to prepare a cross reference table comparing a document before a revision with the document after the revision. It is sometimes necessary to also inspect the content of an optional revision made in a document.

Further, it is necessary to selectively cancel a specific revision (draft withdrawing processing) of multiple revisions that are intricately related to one another or selectively reflect a specific revision in an

original document (assimilation processing), depending on whether or not the revision of a document is acceptable.

The above operation is described in the flow chart in Fig. 1. In Fig. 1, the numerical symbol (11) denotes an original ordinance to be revised, that is, an object of a revision. Performing a revision operation (12) on original ordinance (11), a revision draft (13) is prepared. A relationship between original ordinance (11) and revision draft (13) is denoted by a partially revised ordinance (14), whereas a cross reference table comparing the new version with the old version (15) is prepared. If partially revised ordinance (14) is approved upon deliberations at the Diet, a council meeting or the like, it is promulgated as a revised ordinance (16). Otherwise, an operation for reinstating original ordinance (11) is performed. A new revised ordinance (16) is regarded as an original ordinance at the next revision (17).

Since multiple revision operations are sometimes performed on the same document concurrently as is illustrated in Fig. 2, the aforementioned functions must be implemented without causing any discrepancy among the multiple revisions.

To comply with the above-described requirements, a document revision support system of the present invention constructs, in a storage device of a computer, data expressing a document to be revised as an XML document, in which a volume, a chapter, a clause, an item, a section, a sub-section, a paragraph and the like of a document to be revised are expressed as a tree structure, provides to an element corresponding to a branch node of tree structured data a set of a unique revision ID and an attribute representing a type of the revisions written in a list structure every time a revision is made and saves in the element the original document prior to the revision so that a revision operation of the element is performed using the information.

Hereafter, a structure of such information and an embodiment of such an operation system will be specifically

described by using as an example an operation of revising an ordinance.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a flow chart describing an example of a document revision operation.

Fig. 2 is a flow chart describing how multiple revision operations illustrated in Fig. 1 are performed concurrently.

Fig. 3 is a block diagram of a document revision support system of the present invention.

DETAILED DESCRIPTION OF THE INVENTION

In systematizing processing for generating a revision description, a cross reference table comparing the new version with the old version and a promulgated document, a revision ID for uniquely identifying a location of a revision is provided to each revision location (version) in an ordinance XML (XML data, in which a volume, a chapter, a clause, an item, a section, a sub-section, a paragraph and the like of a law or ordinance to be revised are expressed as a tree structure) with a view to realizing automatization of draft withdrawing processing and assimilation processing on an entire or partial ordinance structure. Once a revision ID is assigned to a revision location, the revision ID will never be changed.

By designating a revision ID, draft withdrawing/assimilation processing is conducted. The processing includes assimilation processing to be conducted for a revision ID subsequent to the designated revision ID and draft withdrawing processing conducted for a revision ID preceding the designated revision ID. Further, as information that forms a set of information with a revision ID, at least one of the following attributes and revision types is added to the document schema (DTD) of the ordinance XML as needed.

(1) revise attribute

A revise attribute is an attribute for adding revise information to a location of a revision. It is a list of sets of a revision ID (version information) representing

edits performed on a document and a revision type representing a type of a revision made to each element. For example, following is a list showing two revisions A and B made to an element X:

5

```
<element X revise="revision A added revision B erased">
```

10 In the above list, the element X represents an element of an ordinance document such as a 'volume', a 'chapter', a 'clause', an 'item', a 'section', a 'subsection', a 'paragraph' and the like, and (revision A added) and (revision B erased) both represent a set of a revision ID and a type of a revision and the order that (revision A added) and (revision B erased) appear in the
15 list corresponds to the order in which the revisions A and B were made. Although the above example concerns an ordinance document, the same as above is applicable to an ordinary document element (as opposed to a legal document element) such as a 'chapter', a 'section', a 'paragraph' and the like and therefore, in the examples described below,
20 a description such as <element X>...</element X> and the like may be optionally replaced by <chapter>...</chapter>, <article>...</article>, <paragraph>...</paragraph> and the like. By describing as illustrated above, it becomes possible to
25 express not only multiple revisions made to an element in the document but also the order in which the revisions are made.

Types of revisions are defined as follows:

- (a) removed: removing an element
- 30 (b) erased: erasing an element. The numbering sequence of any elements which follow is advanced.
- (c) added: adding an element. The numbering sequence of any elements which follow recedes.
- (d) partial: partial revision (revision of a character or a
35 phrase)
- (e) overall: overall revision. The original document is saved.
- (f) revival: An overall revision is made to the removed

elements (almost the same as the above (e)).

(g) collapsed: Effective for a 'volume', a 'chapter', a 'clause', an 'item' and a 'section'. Removing a chapter name or a section name.

- 5 (h) divided: dividing a volume, a chapter, a section, a title, an item and a section.

Next, the aforementioned types of revisions will be respectively described. For each of the types of revisions made to a document or an element in the document, the
10 algorithms for application (assimilation processing) and cancellation (draft withdrawing processing) will be also described.

(a) removed

A revision "removed" means that element X has been
15 "removed". That an element has been "removed" does not mean that the element has been removed from a document but means that the content of the element becomes invalid in the document. For example, this corresponds to the case where the number denoting a specific section is retained
20 (e.g., the numerical symbol '1' in Section. 1 is retained) and the content of the section is marked "removed". (In the specific example given below, a document expressed in XML language above the downward arrow ↓ corresponds to an XML document prior to a revision, whereas one below the
25 downward arrow ↓ corresponds to an XML document after the revision.)

```
<element X removed="no">  
  <element Y>...</element Y>  
30 </element X>  
  ↓  
<element X removed="yes" revise="revision A removed">  
  <element Y>...</element Y>  
</element X>
```

35

A flag in a document that indicates whether the content of an element is valid or not is a removed attribute value.

This indicates that an element which could be removed has a removed attribute.

When a change "removed" (revision A) is applied, "revision A removed" is removed from the revise attribute and further, when the revise attribute value becomes "", the revise attribute is removed from the element X to make the XML document as follows:

```
10  <element X removed="yes">
      <element Y>...</element Y>
    </element X>
```

Thus, the removed attribute becomes yes and the element becomes invalid in the document.

15 On the other hand, when a change "removed" (revision A) is cancelled, "revision A removed" is removed from the revise attribute and make the removed attribute value no.

As is clear from the above, by designating a revision ID, assimilation processing and draft withdrawing processing can be performed in all attributes at once.
(b) erased

25 A revision "erased" means that the element X is "erased". That an element is "erased" means that an element is removed from a document, which corresponds to a case, for example, where not only a specific numerical symbol denoting a section but also the content of the section are removed from a document. In this case, the numbering sequence of any sections that follow the erased section is advanced. A process for advancing the numbering sequence will be described later.

```
30  <element W>...</element W>
      <element X>
        <element Z>...</element Z>
      </element X>
    <element Y>...</element Y>
      ↓
    <element W>...</element W>
```



```
<element X revise="revision A erased">
  <element Z>...</element Z>
</element X>
<element Y>...</element Y>
```

5

When the revision A "erased" is applied, the element X is completely removed from the document to make the XML document as follows:

10 <element W>...</element W>
<element Y>...</element Y>

When the revision A "erased" is cancelled, "revision A erased" is removed from the revise attribute and further, when the revise attribute value becomes "", the revise attribute is removed from the element X.

15

(c) added

A revision "added" means that the element X is added. When the element X is added, the numbering sequence of any sections that follow the "added" section recedes. Processing conducted for such a change in the numbering sequence will be described later.

20

```
<element W>...</element W>
25 <element Y>...</element Y>
      ↓
<element W>...</element W>
<element X revise="revision A added">...</element X>
<element Y>...</element Y>
```

30

When the revision A "added" is applied, "revision A added" is removed from the element X and further, when the revise attribute value becomes "", the revise attribute is removed from the element X to make the XML document as

35 follows:

```
<element W>...</element W>
<element X>...</element X>
```

<element Y>...</element Y>

On the other hand, when the revision A by which the element X was added is cancelled, the element X is completely removed from the document.

(d) partial

A revision "partial" means that a partial revision of a character or phrase is made to an element in which PCDATA (text) and character modifier tags and the like are mixed in an XML document. For example, when a character phrase revision is made to a sentence "It is fine today." to generate a sentence "It's fine today again.", the revision "partial" proceeds as follows:

<element X>It isfinetoday.</element X>

↓

<element X revise="revision A partial">

<del revise_id="revision A">It is<ins
revise_id="revision A">It's</ins>finetoday<ins
revise_id="revision A">again</ins>.
</element X>

In a character phrase revision, an inserted character string is expressed by an ins tag and a removed character string is expressed by a del tag. Further, both tags are respectively provided with a revise_id attribute representing a revision A.

When the revision A by which a character phrase revision was made is applied, "revision A partial" is removed from the revise attribute. Further, the del tag whose revise_id attribute value is "revision A" is removed and a child of the ins tag whose revise_id attribute value is also "revision A" is inserted before the ins tag and the ins tag is removed in the end.

35

<element X revise="revision A partial">

It'sfinetoday again.

</element X>

When the revision A by which a character phrase revision was made is cancelled, "revision A partial" is removed from the revise attribute. Further, the ins tag whose revise_id attribute value is "revision A" is removed and a child of the del tag whose revise_id attribute value is "revision A" is inserted before the del tag and the del tag is removed in the end.

(e) overall

A revision "overall" means that an element is wholly revised. In order to save the former content of the original element prior to the overall revision, the entire former content is saved in the original element unlike a "partial" revision where only a difference between the current content and the former content is saved. A revision ID is saved in a revise_id attribute of an original element. How to revise a full text of an element Z to an element ZZ (a character string of "AAAAA" in this example) will be described below.

```
20 <element X>
    <element Z>...</element Z>
</element X>
    ↓
    <element X revise="revision A overall">
25     <element ZZ>AAAAA</element ZZ>
        <original revise_id="revision A">
            <element X>
                <element Z>...</element Z>
            </element X>
30     </original>
    </element X>
```

When an overall revision is applied, "revision A overall" in the revise attribute of the element X is removed and further, when the revise attribute value becomes "", the revise attribute is removed from the element X and the original element is removed to make the XML document as follows:

```
<element X>
    <element ZZ>AAAAA</element ZZ>
</element X>
```

5 On the other hand, when the overall revision is cancelled, the element X is replaced by one in the original element.

(f) revival

10 A revision "revival" means that a removed element (removed = "yes"), that is, an element that is invalid in the document, is revived. Although "revival" revision processing is basically the same as revision "overall" processing, it is effective only for a removed element. A revision "revival" corresponds to a case where after it is
15 decided when revising an ordinance that a revision "removed" will be conducted on a prevision of the ordinance, it transpires that the provision needs to be revived for some reason.

(g) collapsed

20 A revision "collapsed" means that when elements X1 and X2 of the same kind appear one after another, a division by the element X2 is eliminated. A revision "collapsed" corresponds to a case where a "volume", a "chapter", a "clause", an "item", a "section" and the like
25 are removed.

```
<element X1>
    <element Z1>...</element Z1>
</element X1>
30 <element X2>
    <element Z2>...</element Z2>
</element X2>
    ↓
<element X1>
35 <element Z1>...</element Z1>
</element X1>
<element X2 revise="revision A collapsed">
    <element Z2>...</element Z2>
```

</element X2>

When revision A by which the division by the element X2 is eliminated is applied, a child of the element X2 is transferred to the end of a child of the element X1, which corresponds to its brother, and the element X2 is removed from the document to make the XML document as follows:

```
<element X1>
10   <element Z1>...</element Z1>
      <element Z2>...</element Z2>
</element X1>
```

When revision A by which the division of the element X2 is eliminated is cancelled, "revision A collapsed" is removed from the revise attribute of the element X2 and further, when a revise attribute value becomes "", the revise attribute is eliminated from the element X2.

(h) divided

20 A revision "divided" means that a division of an element is added to thereby divide a child of the element. Revision "divided" processing is a reverse operation of revision "collapsed" processing and corresponds to a case where a "volume", a "chapter", a "clause", an "item", a
25 "section" and the like are added.

```
<element X1>
      <element Z1>...</element Z1>
      <element Z2>...</element Z2>
30 </element X1>
      ↓
      <element X1>
          <element Z1>...</element Z1>
      </element X1>
35 <element X2 revise="revision A divided">
      <element Z2>...</element Z2>
</element X2>
```

When the revision by which a division is added is applied, "revision A divided" is removed from the revise attribute of the element X2.

On the other hand, when the revision by which a
5 division is added is cancelled, a child of the element X2 is transferred to the end of a child of the element X1 corresponding to a brother of the element X2 and the element X2 is removed from the document.

10 (2) num attribute

A num attribute is an attribute representing a numerical symbol denoting a provision of an ordinance and is expressed in the x.y.z form, wherein x is a branch number of a depth 0, y is a branch number of a depth 1 and
15 z is a branch number of a depth 2. More specifically, "2.2.2" is a numerical symbol of a depth 2, "3.3" is a numerical symbol of a depth 1 and "1" is a numerical symbol of a depth 0. A numerical symbol is automatically input or changed such that a discrepancy (redundancy or dropout of
20 numerical symbols) does not arise. By using a dialog designating a location where element Z with a numerical symbol is to be added for designating element X with a numerical symbol, the element Z is inserted after the designated element X. When setting a branch number, a
25 branch number equal to 1 + a numerical symbol representing a depth of a branch number of the element X can be set. In this example (numbering provisions of an ordinance), a branch number of a depth 1 or above starts from 2 (1 is never set). For example, if the branch number of the
30 element X is "5", that of the element Z should start from "5.2" and if the branch number of the element X is "4.2", that of the element Z should start from "4.2.2".

(3) preset attribute

35 A preset attribute is an attribute for processing to carry up/down a number denoting a provision of an ordinance and is expressed as a list of sets of a revision ID (version) representing edits performed on a document and a

numerical symbol of an element in the revision ID.

When a number needs to be carried up or down as a result of a revision, a num attribute value per se representing the number is not changed but the number the
5 original number has been changed to as a result of the revision as well as the order in which they have been changed is controlled for each revision.

A preset attribute is imparted to an element with a numerical symbol (num attribute) and an element whose
10 numerical symbol needs to be carried up/down as a result of removal/addition of another element.

For example, when a new element X11 is added after the element X1 by the revision A made to the following
15 original data:

```
<element X1 num="1">...</element X1>          //1
<element X2 num="2">...</element X1>          //2
<element X3 num="3">...</element X1>          //3
```

20 the following operation is performed on the above data: (a sentence following the symbol // is a comment made solely for making the operation easier to understand).

```
25 <element X1 num="1">...</element X1>          //1
   <element X11 num="2" revise="revision A added">...</element
   X1>          //2 (new)
   <element X2 num="2" preset="revision A 3">...</element X1>
       //2 is carried down to 3
30 <element X3 num="3" preset="revision A 4">...</element X1>
       //3 is carried down to 4
```

Further, when the element X2 is erased by the same revision A, the following operation is performed:

35

```
<element X1 num="1">...</element X1>          //1
<element X11 num="2" revise="revision A added">...</element
X1>          //2 (new):revision A
```

```
<element X2 num="2" preset="revision A 3" revise="revision  
A erased">...</element X1>  
<element X3 num="3">...</element X1>           //4 is carried up  
to 3
```

5

Now that the element X3 is carried back up to 3 from 4 and the numerical symbol 3 is the same as that assigned prior to the revision A, the preset attribute is removed.

On the other hand, when the element X2 is erased by
10 revision B, the following operation is performed.

```
<element X1 num="1">...</elementX1>           //1  
<element X11 num="2" revise="revision A added">...</element  
X1>           //2 (new):revision A  
15 <element X2 num="2" preset="revision A 3" revise="revision  
B erased">...</element X1>  
<element X3 num="3" preset="revision A 4 revision B  
3">...</element X1>           //3 is carried down to 4 (revision  
A), which is then carried up to 3 (revision B)
```

20

When revision A is applied, an element whose preset attribute contains the revision A changes its value to a value of a num attribute to thereby remove "revision A number" from the revise attribute. The above example can

25 be illustrated as follows:

```
<element X1 num="1">...</element X1>           //1  
<element X11 num="2">...</element X1>           //2  
<element X2 num="3" revise="revision B erased">...</element  
30 X1>  
<element X3 num="4" preset="revision B 3">...</element X1>  
           //4 is carried up to 3 (revision B)
```

On the other hand, when the revision B is withdrawn,
35 "revision B number" is removed from the preset attribute.
In this case, the above indicated element X3 becomes as
follows:

<element X3 num="4">...</element X1>

Thus, when an element is added to/erased from elements with numerical symbols denoting a "volume", a "chapter", a "clause", an "item", a "section", a "sub-section" and a "paragraph" and the like, the numbering sequence of volumes, chapters, clauses, items, sections, sub-sections, paragraphs and the like can be advanced or recede by the above-described processing.

However, even though a provision is added as a branch number, it does not necessarily affect the numbering sequence.

For example, when an ordinance originally has Article 1 and Article 2 and Article 1.2 needs to be added, the numerical symbol 2 in Article 2 is not carried down. In this case, "1.2" is set as a num attribute for Article 1.2 and a preset attribute is not imparted. Similarly, when an ordinance originally has Article 1, Article 1.2, Article 1.3 and Article 2, the numerical symbol 2 in Article 2 is not carried up if the Article 1.3 is erased from the original.

Further, as for an element of a provision added by the current revision A, when a numerical symbol is carried up or down as a result of the current revision and another revision, an operation is not performed on a preset attribute but a num attribute value per se is directly manipulated. For example, when a new Article 2 is added between Article 1 and Article 2, data changes as follows:

```
<article num="1">...</article>
<article num="2" revise="revision A added">...</article>
<article num="2" preset="revision A 3">...</article>.
```

If Article 1 is further erased, the following operation is performed:

```
<article num="1" revise="revision A erased">...</article>
<article num="1" revise="revision A added">...</article>
```

//num attribute is changed
<article num="2">...</article> //2 is the same
numerical symbol as that prior to the revision A

5 This is because as the element added by the current
revision does not have "a numerical symbol prior to the
revision" and therefore, there is no "numerical symbol
prior to revision" that needs to be saved, a preset
attribute does not need to be employed.

10 Such a data structure as described above which
employs a set of a revision ID and an attribute makes it
possible to keep multiple revisions independent of one
another and to selectively reflect the content of each
revision on one another.

15 Next, (a) where multiple partial revisions are made
concurrently and (b) where a partial revision and a partial
revision of the partial revision are made concurrently will
be specifically described below.

 First, a provision "Article 1 A is an animal." will
20 be used as an example.

(a) Where multiple partial revisions are made
concurrently

 It is assumed that the following revisions will be
made:

- 25 1. A first provision reflects the content of a partial
revision of February 14.
2. A second provision reflects the content of a partial
revision of March 14.

 Assuming here that as of February 10, data contains
30 the following structure.

<article num="1" revise="r20020214 r20020331">
<p><del revise="r20020214">A<ins
revise="r20020214">B</ins>is<del revise="r20020331">an
35 animal<ins revise="r20020331">mammal</ins>.</p>
</article>

Conducting revision description generation only on an

element having revise="r20020214", "'A' in Article 1 is revised to 'B'." is generated.

At the time of the revision of February 14, the following assimilation processing is performed:

5 When assimilation processing is performed on an element having revise="r20020214", the data becomes as follows:

```
<article num "1" revise="r20020331">  
10 <p>B is an<del revise="r20020331">animal</del><ins  
revise="r20020331">mammal</ins>.</p>  
</article>
```

15 Next, conducting revision description generation only on an element having revised="r20020331", "'animal' in Article 1 is revised to 'mammal'." is generated.

At the time of the revision of March 31, the following assimilation processing is performed. In other words, when assimilation processing is performed on an
20 element having revise="r20020331", the following data is generated:

```
<article num="1">  
<p>B is a mammal.</p>  
25 </article>
```

(b) When a partial revision and a partial revision of the partial revision are made.

It is assumed here that the current data is as
30 follows:

```
<article num="1" revise="r20020214">  
<p><del revise="r20020214">A</del><ins  
revise="r20020214">B</ins>is an animal.</p>  
35 </article>
```

Edits are performed on the above data such that the above data becomes as follows:

```
<article num="1" revise="r20020210 r20020214">
<p><del revise="r20020210">A</del><ins
revise="r20020214"><del revise="r20020210">B</del><ins
revise="r20020210">C</ins></ins>is an animal.</p>
5 </article>
```

This means that it is appropriate to make a partial revision of the partial revision to be made before February 14. Thus, it is assumed that the date is set to be
10 February 10.

Conducting revision description preparation only on an element having revise="r20020210", "'B' in Article 1 is revised to 'C'." is generated.

The above revision description is a correct content
15 of a partial revision of the partial revision.

Thus, after the assimilation processing of February 10, the data becomes as follows:

```
<article num="1" revise="r20020214">
20 <p><del revise="r20020214">A</del><ins
revise="r20020214">C</ins>is an animal.</p>
</article>
```

Conducting revision description generation here, "'A' in Article 1 is revised to 'C'." is generated.
25

The above revision description is also a correct content as a partial revision.

Next, at the time of the revision of February 14, the following assimilation processing is conducted. In other
30 words, performing assimilation processing only on an element having revise="r20020214", the following data is generated as a result of the assimilation processing:

```
<article num="1">
35 <p>C is an animal.</p>
</article>
```

A document revision support system employing the

above data structure is realized by synergy of hardware resources of a computer system and software illustrated in Fig. 3. In Fig. 3, the numerical symbol (30) denotes a document revision supporting device in accordance with the present invention, which comprises: means (31) for inputting a document to be revised; a storage device (32) for storing an input document to be revised and data being processed; means (33) for converting an input document to be revised to data expressed in a tree structure; means (31) for designating an element of stored data and instructing to revise the element; means (34) for generating a list structure consisting of sets of a revision ID uniquely representing a designated revised version and a type of revision to be made to an element and storing the thus generated list structure in the tree structure data; means (35) for saving in the element the original prior to the revision; means (37) for designating a specific revision ID and generating a cross reference table for comparing a revised document or revised portion with the original by using the aforementioned element and attribute; and means (38) for designating a specific revision ID to delete the aforementioned revision ID and attribute and restore the saved document to thereby cancel the revision or delete the revision ID, attribute and saved original document by designating a specific revision ID, and further reflecting a revision represented by the attribute on the original and the like.

Of the aforementioned components, the input/instruction means (31) may be an input/output device such as a keyboard, scanner, voice input device, mouse and the like, and the storage device (32) may be a semiconductor memory device, memory device of a hard disk device and the like, and the converting means (33), generation means (34, 37), saving means (35) and cancel/reflection means (38) and the like are ideally comprised of CPU and various software modules (including firm ware). These components may be realized either on a single computer system or hardware distributed over a

network. For example, a data base server on a network may be employed as the storage device (32), on which an XML document is stored, and the other input/output device and processing means (excluding means (32)) may be realized on
5 a client machine connected to the network.

By implementing the aforementioned document revision support system, the present invention can perform the following functions, which are not provided by the prior art systems.

- 10 1. To automatically generate from a revision record a "revision description" that specifically describes the contents of a revision made at certain point in time.
2. To selectively cancel a specific revision among multiple revisions that are intricately related to one another.
- 15 3. To concurrently perform a multiplicity of revision operations on the same document.
4. To control multiple revisions independently of one another and selectively reflect the contents of the revisions on the original document.
- 20 5. To automatically generate a cross reference table comparing a document before a revision with the document after the revision; and
6. To inspect the content of an optional revision in a document.

25 As is clear from the above, a document revision support system of the present invention is able to satisfy the needs for stringent control of a record of revisions made to, for example, laws, ordinances, legal documents, patent specifications, various manuals, specifications,
30 directories and the like and for strict control of revision histories of editing in the field of publishing and the like.

Although the above embodiment of this present invention has been described as a revision made to an
35 ordinance, those skilled in the art will readily appreciate that the present invention is applicable to various documents. Therefore, it is to be understood that the constitution of the above embodiment is not to be construed

as limiting of the present invention and many modifications that those skilled in the art could think out are intended to be included within the scope of this invention.